**Smart Home Energy Monitor**

Install Raspbian (monitor, keyboard, and mouse)

* Username: **pi** (default)
* Password: **raspberry** (default)

Setup a static IP address (**this step is optional**)

* <https://pihw.wordpress.com/guides/direct-network-connection/> (source)
* **sudo ifconfig eth0 169.254.0.2**
* **hostname -I** (check if it worked)
* Saving the new configuration
  + **sudo cp /boot/cmdline.txt /boot/cmdline.normal** (copy of the original file)
  + **sudo nano /boot/cmdline.txt** (edit the original file)
  + Add at the end of the long line **ip=169.254.0.2** (add a space between the last item and “ip=169.254.0.2”)
  + **crt+x** and **y** (to save and exit)
  + **sudo cp /boot/cmdline.txt /boot/cmdline.direct** (copy of the new file)
  + **sudo reboot** (next time the IP address will be automatically set)

Update Raspbian

* **sudo apt-get update** (fetches the list of available updates)
* **sudo apt-get upgrade** (strictly upgrades the current packages)
* **sudo apt-get dist-upgrade** (installs updates, new ones)

TCP vs UDP

* <http://pymotw.com/2/socket/udp.html> (source)
* TCP (ensuring that all of the data is transmitted in the right order)
* UDP (delivery is not guaranteed, faster than TCP, single packet = **only hold 65,507 bytes**)

PuTTY and WinSCP

* <http://www.putty.org/> (download PuTTY)
* <http://winscp.net/eng/download.php> (download WinSCP)
* The **PuTTY** program is to **use Linux** on another computer using the IP address
* **WinSCP** is another good program to **transfer files** between them

GitHub

* <https://github.com/thiagopuga/Project.git> (project files)

Setup a remote desktop for Raspberry Pi

* <http://www.raspians.com/knowledgebase/?knowledgebase=setting-up-a-remote-desktop-view-the-pi-on-your-windows-pc/> (source)
* Install **Xming** on Windows
  + <http://sourceforge.net/projects/xming/> (download Xming)
* **"C:\Program Files (x86)\Xming\Xming.exe" :0 -clipboard -rootless -screen 0 800x600+100+100@1** (to set window size on the shortcut, +100+100 is the window’s position on the screen)
* Install **PuTTY** on Windows (previous step)
  + Run PuTTY
  + Select **SSH** as the connection type
  + Enter in your Pi’s **IP address** as the Host Name
  + The port should be **22** unless you know better
  + In PuTTY’s option tree, select **Connection/SSH/X11**
  + Check the box labelled **Enable X11 forwarding**
  + Go back to **Session** options (in the option tree)
  + If you would like to save these settings, type a name in the Saved Sessions box and click **Save**
  + Click **Open**
  + Once you have logged into the Pi type **startlxde** and you will see the desktop of your Raspberry in the Xming window (**the Xming must be running on Windows**)

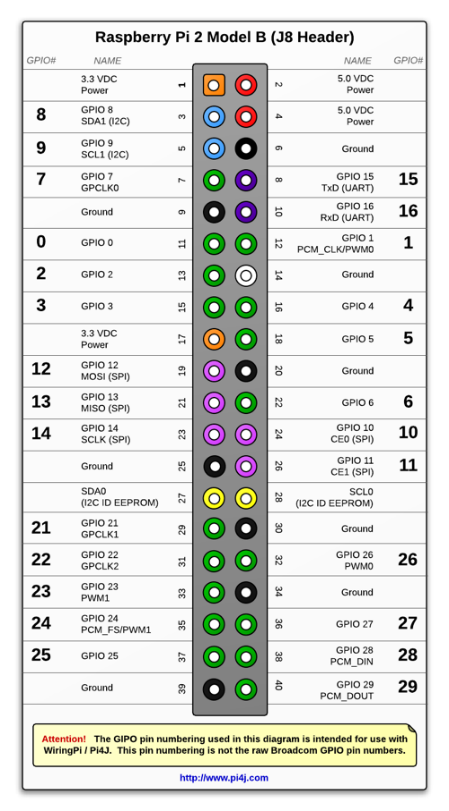
GPS

* <http://blog.mostlyrobots.net/2014/07/07/raspberry-pi-stratum-1-ntp-server/> (source)
* <http://pi4j.com/pins/model-2b-rev1.html> (40-pin Header)
* Connect VIN to +5V
* Connect GPS TX (data out from GPS) to Raspberry Pi RX (data into Raspberry Pi)
* Connect GPS PPS (data out from GPS) to Raspberry Pi PCM\_CLK (GPIO 1, data into Raspberry Pi)
* Connect GND to Ground
* **sudo nano /boot/cmdline.txt**
  + Remove ***console=ttyAMA0,115200*** and save it
* **sudo nano /etc/inittab** 
  + Use **#** to comment out the line ***T0:23:respawn:/sbin/getty -L ttyAMA0 115200 vt100***
* **sudo reboot**
  + **sudo cat /dev/ttyAMA0** (test GPS NMEA)
* Install gpsd
  + **sudo apt-get install gpsd gpsd-clients python-gps**
  + **sudo gpsd /dev/ttyAMA0 -F /var/run/gpsd.sock** (start the communication)
  + **cgps -s** (test if it receives GPS data)
* Configure gpsd to auto start
  + **sudo dpkg-reconfigure gpsd**
  + The configuration program will ask you a series of questions:
    - Start gpsd automatically? **Yes**
    - Should gpsd handle attached USB GPS receivers automatically? **No**
    - Device the GPS receiver is attached to: **/dev/ttyAMA0**
    - Options to gpsd: **-n**
    - gpsd control socket path: **/var/run/gpsd.sock**
  + **sudo reboot**
  + **cgps -s** (you should get the same output as before. this shows everything is starting up correctly on boot)
* Configure NTP
  + **sudo cp /etc/ntp.conf /etc/ntp.old.conf** (make a backup)
  + **sudo nano /etc/ntp.conf** 
    - Use **#** to comment out the line ***restrict -4 default kod notrap nomodify nopeer noquery***
    - Add these lines:  
      ***# Server from shared memory provided by gpsd  
      server 127.127.28.0 minpoll 4 maxpoll 4  
      fudge 127.127.28.0 time1 0.150 refid GPS***
  + **sudo /etc/init.d/ntp restart**
  + **sudo ntpq -p -n**
* Install user mode PPS module
  + <http://vanheusden.com/time/rpi_gpio_ntp/> (source)
  + **sudo wget http://vanheusden.com/time/rpi\_gpio\_ntp/rpi\_gpio\_ntp-1.5.tgz**
  + **sudo tar -zxvf rpi\_gpio\_ntp-1.5.tgz**
  + **cd rpi\_gpio\_ntp-1.5**
  + **sudo make install**
  + **sudo nano /etc/rc.local**
    - Add the following line (**BEFORE the “exit 0”** statement): ***/usr/local/bin/rpi\_gpio\_ntp -N 1 -g 18*** (This assumes that the PPS signal of the GPS is connected to **GPIO pin 18**, which is **physical pin 12**)
    - This program was made for **Raspberry Pi Model B 2**, so we need to adapt which pin to use in our **Raspberry Pi Model B 2 +**
    - [http://www.element14.com/community/docs/DOC-73950/l/raspberry-pi-2-model-b-gpio-40-pin-block-pinout](http://www.element14.com/community/docs/DOC-73950/l/raspberry-pi-2-model-b-gpio-40-pin-block-pinout%20) (**Raspberry Pi Model B** 2 40-pin Header)
  + **sudo rpi\_gpio\_ntp -g 18 -d** (test PPS signal)
* Add the PPS configuration to NTP
  + **sudo vi /etc/ntp.conf**
    - Add these lines:  
      ***# Server from PPS module   
      server 127.127.28.1 minpoll 4 maxpoll 4 prefer  
      fudge 127.127.28.1 refid PPS***
  + **sudo /etc/init.d/ntp restart**
  + **sudo ntpq -p -n**

cgps -s with strange letters (**this step is optional**)

* Open **PuTTY** > **Window** > **Translation**
* Change **UTF-8** to **ISO-8859-1:1998 (Latin-1, West Europe)**

Reading analog-to-digital

* Use a analog-to-digital converter (ADC)
  + MCP3008 VDD -> 3.3V
  + MCP3008 VREF -> 3.3V
  + MCP3008 AGND -> GND
  + MCP3008 CLK -> SCLK
  + MCP3008 DOUT -> MISO
  + MCP3008 DIN -> MOSI
  + MCP3008 CS -> CE0
  + MCP3008 DGND -> GND
* <https://github.com/thiagopuga/Project/tree/master/ADC> (GitHub)

Set NMEA update 10 HZ and NMEA output RMC only

* Connect GPS RX (data out from GPS) to Raspberry Pi TX (data into Raspberry Pi)
* Run the **SendCommandGPS.py** (code)

MySQL

* **sudo apt-get install mysql-client**
* **sudo apt-get install python-mysqldb**

Amazon Web Services‎ (AWS)

* <http://aws.amazon.com/> (link)
* Email: **s3laws@gmail.com**
* Password: **s3ls3ls3l**
* Configuring the server
  + <http://blog.clearpathsg.com/blog/bid/343084/Creating-a-MySQL-Instance-Using-AWS-RDS-and-Accessing-it-Using-MySQL-client> (source)
  + MySQL
    - User: **awsuser**
    - Password: **MyDatabase**
    - Database: **SPI**
    - **mysql -hmydbinstance.cmkub5asq0w1.us-west-2.rds.amazonaws.com -P3306 -uawsuser -pMyDatabase** (login example)